EXHIBIT D

APPLICATION FOR PATENT UNDER 37 C.F.R. §1.53(c) TRANSMITTAL FORM

COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, VA 22313-1450

Attorney Docket No. **076533.0130**

Dear Sir:

Enclosed application parts are: __X__ Transmittal Form Total number of Pages __1__
__X__ Spec w/Claims Total number of Pages __41__
__X__ Drawings Total number of Pages __4__

Inventor(s):

LAST NAME	FIRST NAME	MIDDLE INITIAL	RESIDENCE (CITY, STATE OR FOREIGN COUNTRY)
Waeschenfelder	Robert		Dallas, Texas
Ibarrola	Germinal		Fairview, Texas
Roberts	Anna	M.	Dallas, Texas
Carrico	Todd	M.	Sachse, Texas
Stone	James	C.	Addison, Texas
Van Sligtenhorst	Anna	L.	Murphy, Texas

TITLE OF INVENTION: **MATCHING PROCESS SYSTEM AND METHOD**

CORRESPONDENCE ADDRESS: Chad C. Walters
PHONE: (214) 953.6511 FAX: (214) 661.4511

__X__ Customer Number **05073**

Was this invention made under a Government contract? __X__ No ___ Yes
Identify contract and the Government agency: N/A

The Commissioner is hereby authorized to charge the Provisional Application Filing Fee in the amount of $105.00 for a small entity filing fee to the Deposit Account No. 02-0384 of Baker Botts L.L.P. The Commissioner is also authorized to charge any extra fees or credit any overpayment to Deposit Account No. 02-0384 of Baker Botts L.L.P.

Respectfully submitted,

Date: December 19, 2007

Chad C. Walters
Registration No. 48,022

PROVISIONAL APPLICATION FOR PATENT ONLY SMALL ENTITY

MATCHING PROCESS SYSTEM AND METHOD

TECHNICAL FIELD

This invention relates generally to computer systems and more particularly to a matching process system and method.

BACKGROUND

Networking architectures have grown increasingly complex in communications environments. In recent years, a series of protocols and configurations have been developed in order to accommodate a diverse group of end users having various networking needs. Many of these architectures have gained significant notoriety because they can offer the benefits of automation, convenience, management, and enhanced consumer selections.

Certain network protocols may be used in order to allow an end user to conduct an on-line search of candidates to fill a given vacancy. These protocols may relate to job searches, person finding services, real estate searches, or on-line dating. While some believe that on-line dating is simply a matter of matching supply and demand, there is statistical and empirical evidence to suggest that successful on-line dating entails far more.

For example, people having similar and/or compatible character traits and values should be matched together. However, effectively linking two participants together can prove to be a challenging endeavor. Coordinating a relationship between two like-minded individuals can be a significant chore, as there are a number of obstacles and barriers that must be overcome.

One problem that has arisen is that matching services are limited to searching for matches only within their own platform. Thus, only people who have gone through the process of signing up for the service are searched for a match. One solution to this problem is to have users register in multiple services. This is problematic because it can be expensive and time

consuming for users. Further, the user must then visit all of the services to monitor the search progress: this inefficiency may cause users to give up on the search process.

Another problem is that the search results of these services contain many irrelevant entities to the searcher. This costs the user of the service time and may deter them from continuing through all of the search results.

SUMMARY

In one embodiment, a matching method comprises generating a pool of entities and applying at least one scoring algorithm to the pool of entities. It also comprises tabulating the at least one score from the at least one scoring algorithm and applying an ordering algorithm to the at least one score. It further comprises presenting a list of matches which are ordered based on the results of the ordering algorithm.

Generating a pool may comprise choosing at least one seed entity and generating a score by comparing the at least one seed entity with other entities. It may further comprise setting a threshold score and adding entities to the pool whose score is above the threshold score.

The matching method may further comprise presenting a list of matches, which are associated with a first user, to a second user. Further, it may comprise the second user indicating a preference for at least one entity in the list of matches associated with the first user. It also may comprise presenting the at least one preference of the second user to the first user.

In another embodiment, a matching system comprises a first platform configured to store a first set of user profiles. It further comprises a second platform configured to store a second set of user profiles. The second platform is further configured to search through both the first set and second set of user profiles. It is also configured to, within the second platform, present to the second platform search results from the first set of profiles. Further, it is configured to,

within the first platform, present to the first platform search results from the second set of user profiles.

Depending on the specific features implemented, particular embodiments may exhibit some, none, or all of the following technical advantages. Various embodiments may be capable of searching for matches on multiple platforms. Some embodiments may be capable of adjusting the search algorithms based on the sex of the user requesting the search for a match. Various embodiments may be capable of updating returned search results based on activities communicated to the system. Other technical advantages will be readily apparent to one skilled in the art from the following figures, description and claims.

BRIEF DESCRIPTION OF THE DRAWINGS

Reference is now made to the following description taken in conjunction with the accompanying drawings, wherein like reference numbers represent like parts, and which:

FIGURE 1A is an overview of one embodiment of the matching system.

FIGURE 1B shows the contents of the terminal from FIGURE 1A.

FIGURE 1C shows the contents of the matching server from FIGURE 1A.

FIGURE 1D is a diagram of a database from FIGURE 1C showing one embodiment of how a matching server stores a pool.

FIGURE 1E is a diagram of the display from FIGURE 1B showing one embodiment of the presentation of search results to a user.

FIGURE 1F is a diagram of the display from FIGURE 1B showing one embodiment of the presentation of details of a match result entity to a user.

FIGURE 2 is a diagram depicting how a user may recommend an entity to another user, in accordance with a particular embodiment.

FIGURE 3 is a diagram of the display from FIGURE 1B depicting how the user may be made aware of fate characteristics the user shares with a match result entity, in accordance with a particular embodiment.

FIGURE 4 is a diagram depicting how two platforms may be searched for a match, in accordance with a particular embodiment.

FIGURE 5 is a flow chart indicating how a result list may be generated, in accordance with a particular embodiment.

DETAILED DESCRIPTION

Referring to FIGURE 1A, one embodiment of a matching system is shown. FIGURE 1A is a simplified block diagram of a system 100 for facilitating an on-line dating scenario in a network environment. In other embodiments, system 100 can be leveraged to identify and to evaluate suitable candidates in other areas (e.g. hiring/employment, recruiting, real estate, general person searches, etc.). Users 14 interact with a matching server 20 through terminals 10. FIGURE 1B is a diagram showing, in one embodiment, the contents of terminal 10. Terminal 10 comprises interface 16 (so that user 14 may be able to interact with terminal 10) and display 12. FIGURE 1C is a diagram showing, in one embodiment, the contents of matching server 20. Matching server 20 comprises memory 26 and at least one CPU 28. Memory 26 may store multiple databases, such as databases 26a and 26b. Terminal 10 and matching server 20 are communicatively coupled via network connections 22 and network 24.

Users 14 are clients, customers, prospective customers, or entities wishing to participate in an on-line dating scenario and/or to view information associated with other participants in the system. Users 14 may also seek to access or to initiate a communication with other users that may be delivered via network 24. Users 14 may review data (such as profiles, for example) associated with other users in order to make matching decisions or elections. Data, as used herein, refers to any type of numeric, voice, video, text, or script data, or any other suitable information in any appropriate

format that may be communicated from one point to another.

In one embodiment, terminal 10 represents (and is inclusive of) a personal computer that may be used to access network 24. Alternatively, terminal 10 may be representative of a cellular telephone, an electronic notebook, a laptop, a personal digital assistant (PDA), or any other suitable device (wireless or otherwise: some of which can perform web browsing), component, or element capable of accessing one or more elements within system 100. Interface 16, which may be provided in conjunction with the items listed above, may further comprise any suitable interface for a human user such as a video camera, a microphone, a keyboard, a mouse, or any other appropriate equipment according to particular configurations and arrangements. In addition, interface 16 may be a unique element designed specifically for communications involving system 100. Such an element may be fabricated or produced specifically for matching applications involving a user.

Display 12, in one embodiment, is a computer monitor. Alternatively, display 12 may be a projector, speaker, or other device that allows user 14 to appreciate information that system 100 transmits.

Network 24 is a communicative platform operable to exchange data or information emanating from user 14. Network 24 could be a plain old telephone system (POTS). Transmission of information emanating from the user may be assisted by management associated with matching server 20 or manually keyed into a telephone or other suitable electronic equipment. In other embodiments, network 24 could be any packet data network offering a

communications interface or exchange between any two nodes in system 100. Network 24 may alternatively be any local area network (LAN), metropolitan area network (MAN), wide area network (WAN), wireless local area network (WLAN), virtual private network (VPN), intranet, or any other appropriate architecture or system that facilitates communications in a network or telephonic environment, including a combination of any networks or systems described above. In various embodiments, network connections 22 may include, but are not limited to, wired and/or wireless mediums which may be provisioned with routers and firewalls.

Matching server 20 is operable to receive and to communicate information to terminal 10. In some embodiments, matching server 20 may comprise a plurality of servers or other equipment, each performing different or the same functions in order to receive and communicate information to terminal 10. Matching server 20 may include software and/or algorithms to achieve the operations for processing, communicating, delivering, gathering, uploading, maintaining, and/or generally managing data, as described herein. Alternatively, such operations and techniques may be achieved by any suitable hardware, component, device, application specific integrated circuit (ASIC), additional software, field programmable gate array (FPGA), server, processor, algorithm, erasable programmable ROM (EPROM), electrically erasable programmable ROM (EEPROM), or any other suitable object that is operable to facilitate such operations.

In some embodiments, user 14, using terminal 10, registers with matching server 20. Registration may

include user 14 submitting information to matching server 20 about user 14 as well as characteristics user 14 is seeking to be matched with. Such information may include a user handle, which may be a combination of characters that uniquely identifies user 14 to matching server 20. In various embodiments, matching server 20 may be configured to collect this information; for example, matching server 20 may be configured to ask user 14 to respond to a series of questions. Matching server 20 may be configured to receive the information submitted by user 14 and create a profile for user 14 based on that information, storing the profile in memory 26.

As an example only, consider a case where user 14 is interested in participating in an on-line dating scenario. User 14 can access the Internet via terminal 10, travel to a web site managed by matching server 20, and begin the registration process. As part of the registration process, matching server 20 may ask user 14 a series of questions which identifies characteristics about user 14. Thus, matching server 20 may ask about the height, weight, age, location, and ethnicity of user 14. It may also ask about the birthplace, parents, eating habits, activities, and goals of user 14. Matching server 20 may further use the registration process to discover what user 14 may be looking for in a match, such as age, weight, height, location, ethnicity, diet, education, etc. Further, matching server 20 may ask user 14 to indicate how important certain factors are when looking for a match. For example, matching server 20 may allow the user to indicate which characteristics in a potential match are a necessity. In another example, matching server 20 may ask, "How important is it

that your match does not smoke?" Matching server 20 may also allow the user to indicate that certain characteristics are not important search criteria. For example, when asking user 14 about what height or weight user 14 is seeking in a match, matching server 20 may be configured to receive "not important" as a response. In yet another example, matching server 20 may allow user 14 to rate which factors are important on a numerical scale. For example, matching server 20 may ask user 14 the following: "On a scale of 1 - 10, how important is it that your match has the same education level as you?" In some embodiments, matching server 20 may specify that any number of questions or requested descriptions are necessary before registration may be concluded. As an example only, matching server 20 may require that user 14 communicate the sex of user 14 and the sex user 14 prefers to be matched with. After concluding the registration process, matching server 20 may store the responses of user 14 as a profile. This same process may be repeated by several different users 14, causing matching server 20 to contain a plurality of profiles.

FIGURE 1D depicts an embodiment in which matching server 20 has a database 26a which contains a pool 30. Each entry in database 26a has a pool entity 30a along with information concerning that entity. In one embodiment, each pool entity 30a-e represents a user and their profile. In some embodiments, not all registered users are in pool 30. As discussed further below, matching server 20 may use a selection process for including stored profiles in pool 30. As depicted in FIGURE 1D, in this embodiment, the collection of users and profiles forms pool 30 through which matching server

20 may perform various functions such as searches for matches.

Matching server 20 may be configured to search through pool 30 and present matches to user 14. In FIGURE 1E, one embodiment of this presentation is depicted as occurring through display 12. In this embodiment, a result list 31 is presented to user 14. A match result entity 31a in a result list 31 may be associated with a view button 33. Using interface 16, user 14 may request that matching server 20 provide more information about an entity in result list 31 by pressing the associated view button 33. Matching server 20 may then communicate to user 14 more information about that entity by retrieving the information from memory 26. In FIGURE 1F, one embodiment of information that matching server 20 provides for user 14 is shown. Using display 12, user 14 views an entity from result list 31. Matching server 20 may also provide user 14 with the ability to contact the entity through a contact button 35. In one embodiment, when contact button 35 is utilized by user 14, matching server 20 may provide user 14 with contact information of the entity such as a telephone number or an e-mail address; in another embodiment, matching server 20 may provide user 14 with a way to directly contact the entity, such as sending a message or providing voice or video communication between user 14 and the entity. Even further, matching server 20 may be configured to allow user 14 to express a negative preference for the entity through dislike button 36. In one embodiment, when, for example, dislike button 36 is utilized by user 14, matching server 20 may remove the entity from result list 31; in another embodiment, the

entity may be removed from pool 30 of users from which matches are identified.

As an example only, consider that user 14 has submitted a search request to matching server 20. Matching server 20 may search through pool 30, identify results, and communicate result list 31 to user 14 which would contain other users for whom matching server 20 had created a profile and who were identified through a search and selection process. Next, user 14 may be interested in learning more about Jane Doe, entity 31a; thus, user 14 would click view button 33 associated with Jane Doe. Matching server 20 would receive this request and respond by displaying Jane Doe's profile (stored in memory 26), as depicted in FIGURE 1F. Next, after reading the profile, user 14 may be interested in contacting Jane Doe; hence, user 14 would click contact button 35. Matching server 20 would respond by allowing user 14 enter a message that matching server 20 would then communicate to Jane Doe.

Matching server 20 may even further be configured to allow user 14 to store a match result entity; in one embodiment, the system may be configured to allow user 14 to utilize favorite button 34 that will add the desired match result entity into another list. In another embodiment, utilizing favorite button 34 will remove the associated match result entity from result list 31.

As an example only, user 14 may decide that he would like to save Jane Doe's profile so that he can review it later. User 14 may click favorite button 34, and matching server 20 may respond by placing Jane Doe's profile into a separate list. Further, matching server 20 may also remove Jane Doe from user's 14 result list

31. As a result, user 14 may see another match result entity populate result list 31. This is beneficial because it may focus user 14 on evaluating new entities rather than reevaluating previously-known entities because the entities still appear in result list 31.

In some embodiments, matching server 20 may be configured to generate pool 30 by default according to various characteristics and preferences of user 14 and other users of the system. Matching server 20 may also restrict entities from being included in pool 30 based on the status of the profile, or if user 14 has rejected or blocked an entity. Matching server 20 may also restrict entities from the pool that have blocked or rejected user 14. For example, matching server 20 may not allow profiles that are not in good standing to be included in pool 30. In other embodiments, matching server 20 may be configured to generate pool 30 by first choosing seeds. Seeds include, but are not limited to, profiles that user 14 has sent a message to or profiles that user 14 has expressed a preference for. Each seed is then compared to other entities to determine which entities will be included in pool 30. Any suitable method can be used to determine which entities are included in pool 30. For example, any characteristics or algorithms described herein may form the basis of such a determination. As another example, a commonality score may be generated based on the comparison between each entity and the seed. In some embodiments, this commonality score can be a measure of how physically similar the users are to each other. This score may be generated based on the number of users that have expressed a positive preference for both the seed and the entity being compared. This score

may also be generated based on whether the seed and entity have been viewed together in one session; further, the more times the seed and entity have been viewed together, the larger the commonality score. The law of large numbers may allow for a vast amount of such commonalities to be established over a few days. Testing has revealed that using such commonality scoring methods has yielded at least one physical match for 80% of users whose profile has been viewed at least once, and between 6 and 1000 physical matches for 60% of users whose profile has been viewed at least once. Matching server 20 may be further configured to allow entities that have a commonality score above a certain threshold to become a part of pool 30. Matching server 20 may further be configured to update pool 30. In some embodiments, matching server 20 may do so by creating new seed entities based on activity by user 14, such as indicating a preference for that entity. Further, matching server 20 may then compare the chosen seed entity with other profiles stored in matching server 20 and determine whether those profiles will be included in pool 30 using a threshold score as described above. At least one advantage realized by this embodiment is that user 14 is presented with updated potential matches which increases the likelihood of user 14 finding a suitable match. Another advantage present in certain embodiments is that these updated potential matches have a greater likelihood of compatibility with user 14 since they are chosen based on their commonality with entities user 14 has expressed a preference for.

As an example only, consider the case in which user 14 has registered, requested a search, and received from

matching server 20 results list 31. Then, user 14 decides to contact Jane Doe and presses contact button 35. Aside from providing user 14 with the ability to contact Jane Doe, matching server 20 will designate Jane Doe's profile as a seed. Matching server 20 will then compare Jane Doe's profile to other profiles stored in memory 26 in order to identify other users who may be similar to Jane Doe and thus be a good match for user 14. In this example, matching server 20 will generate a commonality score for each of these comparisons and compare these scores to a preset threshold. If the commonality score is lower than the threshold, that profile will not be added to pool 30. However, if the commonality score is higher than the threshold, matching server 20 will add this profile to pool 30. As an example, further assume that the seed, Jane Doe, is being compared to another entity, Susan Smith. Based on the fact that both Susan and Jane have three users (Tom, Dick, and Harry) who have expressed a positive preference for their profiles, matching server 20 generates a commonality score of 100 for the comparison. In contrast, matching server 20 generated a commonality score of 50 for the comparison between the seed (Jane Doe) and yet another entity, Lucy Goosey. This was because only one user (Bob) had indicated a positive preference toward both Lucy and Jane. Continuing the example, matching server 20 is using a commonality threshold score of 70, which results in including Susan's profile (whose commonality score was greater than the threshold score) in pool 30 and excluding Lucy's (whose commonality score was less than the threshold score).

Thus, user 14 gets the benefit of having more entities identified that may be good matches.

In some embodiments, matching server 20 may be configured to include behavioral scales. These may include multi-item scales for materialism and gender-role traditionalism. Such scales may provide the advantage of improved matching through deeper appreciation for the personality of entities in the system.

In some embodiments, matching server 20 may be configured to analyze profile text for categories. It may search for a number of text strings and then associate the profile with any number of categories. As an example only, matching server 20 may add any profile to the Cat category whose text contains any of the following strings:

"cat" "cats" "cat." "cats." "cat," "cats,"

Matching server 20 may be configured to make it more likely that a profile will be in a result list if categories associated with the profile are also categories found in the user's profile who submitted the search request.

Matching server 20 may be configured to analyze one or more portions of the text of an entity's profile and generate a readability score that may be used in various ways, such as in the process of searching for matches for user 14. In some embodiments, matching server 20 may analyze factors such as, but not limited to: average number of words per sentence, total number of words with greater than three syllables, and total number of words in the profile. Matching server 20 may also concatenate

all of the collected responses with a single space between them. It may further break the text into sentences, words, and syllables. From these statistics, matching server 20 may also be configured to generate a readability score by, in one embodiment, taking the average of the Flesch Kincaid Reading Ease test, the Flesch Kincaid Grade Level test, and the Gunning Fox score. Other embodiments may utilize any other combination of these or other tests to determine a readability score. In some embodiments, analyses may be used to determine the IQ of an entity, the grade level of the writing, or how nervous the entity generally is. An advantage of this embodiment may be that the system provides user 14 with a metric for determining approximate intelligence of other users. The readability score may be used, for example, in the matching process to identify potential matches.

As an example only, the Flesch Kincaid Reading Ease score may be generated by first computing the following intermediate score:

206.835 - (1.015 * [Average Words per Sentence]) - (84.6 * [Average Syllables per Word])

Then, the Flesch Kincaid Reading Ease score is determined by using the following table:

Intermediate Score Condition	Flesh Kincaid Reading Ease Score
< 100	4
< 91	5

< 81	6
< 71	7
< 66	8
< 61	9
< 51	10
< 31	13
< 0	14
Else	15

The Flesch Kincaid Grade Level may be computed according to the following:

(0.39 * [Average Words Per Sentence]) + (11.8 * [Average Syllables Per Word]) - 15.59

The Gunning Fox score may be computed according to the following:

([Average Words Per Sentence] + (([Number Of Words With More Than 3 Syllables] / [Number of Words In Entire Text]) + 100)) * 0.4

As indicated, any suitable tests may be utilized in any suitable manner to determine a readability score.

In some embodiments, matching server 20 may be configured to allow a user to interact with the result list of another user. Matching server 20 may be configured to allow a user to express a preference for entities within a result list of another user, and to indicate to the other user of this preference. Thus, a user may be able to get advice from a friend regarding

what other users may constitute good matches for the user and thus be able to find a better match.

As an example only, consider FIGURES 1A and FIGURE 2. Two users 14, Harry and Sally, are connected to matching server 20 via terminals 10. Display 12a is used by Harry while display 12b is used by Sally. Matching server 20 allows Sally to view Harry's result list 31 on her terminal in display 12b. By pressing recommend button 37, Sally may indicate a preference for one or more of the entities in result list 31. Assume Sally presses recommend button 37 associated with Jane Loe. After doing so, matching server 20 will notify Harry of Sally's preference. On Harry's display 12a, matching server 20 will cause notification 39 to appear, associating it with Jane Loe. Notification 39 will indicate to Harry that Sally has recommended Jane Loe as a potential match. Harry may find Sally's preference helpful in determining which entities he should pursue further if, for example, he believes Sally understands the type of person he is looking for.

In one embodiment, matching server 20 may be configured to analyze the profiles of both user 14 and the entities in pool 30 for keywords. Matching server 20 may be configured to search through the profile of user 14 for keywords that relate to things such as activities and interests. Matching server 20 may generate a score for each entity in pool 30 based on a comparison between the list of keywords found in user's 14 profile and a similarly-generated list of keywords of each entity in pool 30. In one embodiment, this is accomplished by storing a list of words in memory 26, and using it to identify keywords in the searched profiles. In some

embodiments, identified keywords may be used as a means of weighting various scores. As an example only, a profile that contains the word "God" may be weighted much differently than a profile which has merely indicated that their religious preference is Christian. In various embodiments, this may provide an advantage to user 14 in that user 14 is able to determine how similar he/she is with a potential match. In addition, the keyword analysis may be used by the system when searching and identifying matches for a user.

As an example only, consider two registered users, Harry and Sally, both of whom have profiles stored in matching server 20. Matching server 20 then analyzes each of these profiles by comparing it to a list of predefined keywords. Matching server 20 then associates each word that matched the list of keywords with each profile. Now assume that Harry performs a search. While fulfilling Harry's query, matching server 20 evaluates Sally's profile for inclusion in Harry's result list 31. This evaluation includes comparing the list of keywords found in Harry's profile to the keywords found in Sally's profile. The more keywords that Harry and Sally have in common, the more likely it will be that matching server 20 will include Sally's profile in Harry's result list 31.

In some embodiments, matching server 20 may be configured to impute a level of physical attractiveness to an entity in pool 30. Matching server 20 may be configured to monitor how frequent an entity in pool 30 has been viewed as well as how many times that entity has been part of a result list in order to impute the level of physical attractiveness. Matching server 20 may

further be configured to generate a score based on this data. Further, in some embodiments, matching server 20 may impute physical attractiveness to an entity based on the imputed physical attractiveness scores of other entities. Matching server 20 may compute an average of the imputed physical attractiveness scores of the other entities weighted by the commonality score between each of the other entities and the present entity. Empirical data indicates that people are more likely to match with people of similar attractiveness. Thus, in many embodiments, a user may obtain an advantage in that they are able to be presented with potential matches that, according to one measurement, are as attractive as the user.

As an example only, consider a registered user, Sally, whose profile was created by matching server 20 in January. Since that time, matching server 20 has recorded the number of times Sally's profile has appeared in any user's result list 31; assume that this has occurred 10 times. Further, matching server 20 has also recorded the number of times a user has viewed Sally's profile by clicking view button 33 associated with Sally's profile; assume that this has happened 5 times. In this manner, matching server 20 has constructed a ratio that represents the imputed physical attractiveness of Sally's profile. Still further, assume that Harry, a registered user, now submits a query. Matching server 20 has evaluated the imputed physical attractiveness ratio of Harry's profile. When evaluating Sally's profile for inclusion in result list 31 returned to Harry, matching server 20 will compare the imputed physical attractiveness of Sally's profile and Harry's profile.

The more similar the ratios associated with Harry and Sally's profiles are to each other, the more likely it is that Sally's profile will be selected by matching server 20 to be in Harry's result list 31. In another example, assume that Sally's profile has not been registered long enough to generate a meaningful imputed physical attractiveness ratio. Matching server 20 may then generate an imputed physical attractiveness score based on entities that Sally does have commonality scores with. This computed average may be weighted by the strength of the commonality score between Sally and each entity with whom she has a commonality score. Continuing the example, assume that Sally has a commonality score of 5 with Lucy and 10 with Julia. When matching server 20 computes the Sally's average, it will give twice as much weight to Julia's imputed physical attractiveness score than to Lucy's.

In some embodiments, matching server 20 may be configured to make an entity in result list 31 more appealing to user 14 by pointing out coincidences in the profile data that give user 14 a sense of fate with the entity. In one embodiment, matching server 20 may be configured to search for similar initials, birthplaces, birth dates, birth month, birth year, university, first names, last names, user handles, parental occupations, and keywords to identify users who may give another user a sense of fate. In other embodiments, matching server 20 may use the fate characteristics as a metric in the matching process.

As an example only, assume that Harry is a registered user who has performed a search. After matching server 20 returns a result list, Harry chooses

to learn more about one of the entities in the result list and clicks view button 33. Consider FIGURE 3, which is only an example of information that matching server 20 may return to Harry after clicking view button 33. In Harry's display 12, matching server 20 presents certain details about the profile. In particular, matching server 20 presents to Harry a fate notification 32 which points out specific similarities between the profile of the entity and Harry's profile. Reading fate notification 32 gives Harry a sense of familiarity which enhances his appreciation for the profile.

In another example, fate characteristics may be used to decide whether a profile in pool 30 is included in user's 14 result list 31. Assume that Harry is a registered user who has submitted a matching query to matching server 20. While determining which entities to include in Harry's result list, matching server 20 considers two profiles: Sally and Roxy. Sally and Harry both have the same birth date, initials, and have parents that work in the same profession. In contrast, Roxy and Harry only share the same birth place. Matching server 20 may be configured to award more points to Sally than to Roxy based on these comparisons, making it more likely that Sally's profile will be included in Harry's result list.

In some embodiments, matching server 20 may be configured to evaluate the likelihood of contact between user 14 and an entity in pool 30. Matching server 20 may be configured to compare demographic data between user 14 and pool entity 30a. In another embodiment, matching server 20 may be configured to weigh the demographic similarities and differences based on the sex of user 14.

The demographic data may include, but is not limited to, age, education, ethnicity, income, and location.

As an example only, assume that Harry and Sally are registered users who have profiles in matching server 20. Harry has submitted a search request to matching server 20. While fulfilling this request, matching server 20 evaluates Sally's profile since her profile is in pool 30. As part of the evaluation, matching server 20 looks at the differences between Harry and Sally's stated age, income, education, ethnicity, and location. In this example, Harry is 10 years older than Sally, makes $10,000 more per year, and has a Master's degree while Sally has a bachelor's degree. Even with these disparities, matching server 20 will give Sally's profile a high score which makes it more likely that Sally's profile will appear in Harry's result list. However, if it was Sally who submitted the search, and matching server 20 was evaluating Harry's profile, a different score is possible. So, if it were Sally who was 10 years older, made $10,000 more per year, and had a Master's degree while Harry had a Bachelor's degree, matching server 20 would give a low score to Harry's profile, making it less likely that his profile would appear in Sally's result list. Matching server 20 may be configured this way because empirical data has shown that these demographic differences do not have an equivalent effect on the choices men and women make regarding matches.

In another embodiment, matching server 20 may be configured to compare the locations of user 14 and pool entity 30a in increments of ten miles. In yet another embodiment, matching server 20 may be configured to score

the location comparison in light of other factors; as an example, matching system 20 may be configured to return a score consistent with a 10 mile difference in location even though there is a 50 mile difference between user 14 and pool entity 30a if user 14 and pool entity 30a have the same income, education, and age. An advantage realized in several embodiments is that it better approximates how a user evaluates entities. Entities that live further away are generally less appealing to a user; but, users may still be interested if the entity matches their preferences in other categories.

As an example only, consider a registered user, Harry, who submits a search request. While fulfilling this request, matching server 20 examines Sally's profile in pool 30, and determines that the stated locations of Harry's and Sally's profiles are 13 miles apart. Matching server 20 will give Sally's profile a score as if the distance between them were only 10 miles. However, in yet another example, Sally's profile may indicate that she lives 50 miles away from Harry. Yet, matching server 20 also notes that both Harry and Sally make $100,000 per year, have Master's degrees, and that Harry and Sally are one year apart in age (Harry is older). Given these similarities, matching server 20 will give a score to Sally's profile that is consistent with a 20 mile difference in location even though they are actually 50 miles apart. In this manner, matching server 20 takes into account empirical data that shows that people searching for matches who indicate that they want to see matches who live close to them are still willing to pursue a potential match that lives far away

if the potential match fits very closely with the other search criteria.

In another embodiment, matching server 20 may be configured to evaluate the age difference between user 14 and pool entity 30a using ranges as well as a sliding scale. By way of example only, matching server 20 may be configured to assign a high value to an age difference between 0 and -5, while assigning a lower value to an age difference between +2 and 0. An even lower value may be assigned to an age difference between -6 and -8. Even lower values would be assigned incrementally as the age difference increases outside of the ranges discussed. The higher the assigned value is, the more likely it will be that pool entity 30a will be included in result list 31. Yet another embodiment may apply this combination of ranges and a sliding scale but use different values and ranges depending on the sex of user 14.

As an example only, consider a situation in which a registered user, Harry, requests a search to be performed. While fulfilling this request, matching server 20 evaluates Sally's profile, which was in pool 30. As part of the evaluation, matching server 20 compares the ages of Harry and Sally, and determines that Harry is two years older than Sally; this determination leads to matching server 20 assigning, in this example, 50 points to Sally's profile. Matching server 20 may also be configured to assign 50 points to Sally's profile had she been five years younger than Harry; but, if she had been up to two years older than Harry, matching server 20 may have been configured to assign 40 points to her profile. Matching server 20 may be further configured to assign 30 points to Sally's profile if she

was 6 to 8 years younger than Harry. However, if Sally were more than 8 years younger than Harry, matching server 20 may be configured to further decrease the number of points assigned to her profile: if she was 9 years younger, then 25 points; if she was 10 years younger, 20 points; if she was 11 years younger, 15 points; etc. The more points assigned to Sally's profile, the more likely it is that her profile will appear in Harry's result list. Thus, matching server 20 may be configured to assign a score based on age difference using a combination of ranges and a sliding scale.

In another example, matching server 20 may assign scores differently if it was Sally who was searching and if it was Harry's profile that was being evaluated. In this example, matching server 20 may be configured to assign Harry's profile 50 points if he were between 1 and 5 years older than her. If he were 6 to 8 years older than her, matching server 20 may assign 45 points. If he were greater than 8 years older than her, matching server 20 may assign points in the following fashion: if he was 9 years older, 40 points would be assigned; if he was 10 years older, 35 points would be assigned; etc. However, if he was up to two years younger than Sally, matching server 20 may assign 50 points to his profile. If he were more than two years younger, matching server 20 may assign less points on a sliding scale: 45 points if he were 3 years younger, 40 points if he were 4 years younger, etc. The more points assigned to Harry's profile, the more like it is that his profile will appear in Sally's result list. This example illustrates how matching server 20 may be configured to take the sex of

user 14 into account when scoring based on age differences.

In various embodiments, matching server 20 may be configured to evaluate the attractiveness of an entity in pool 30 through collected feedback from other users. In one embodiment, matching server 20 may present an entity to user 14, prompting user 14 to rate the attractiveness of the entity on a scale from 1 - 9. This range gives the advantage of having a midpoint. Matching server 20 may further be configured to collect such responses and store them; in one embodiment, matching server 20 may store the data in memory 26, using a structure such as database 26b. Matching server 20 may further be configured to compute the average of such responses for the entity, and store this number as well. In various embodiments, these values may be used in order to help in the matching process. Empirical data indicates that people are more likely to match with people of similar attractiveness. Thus, in various embodiments, users whose attractiveness rating are similar will be more likely to appear in each other's result list. Further, a user may indicate that they only want profiles in their result list whose average attractiveness rating is higher than an indicated threshold.

As an example only, assume registered user, Harry, uses terminal 10, which in this example is Harry's personal computer, and establishes communication with matching server 20. In this example, this communication occurs by Harry using a Web browser to access a Web page controlled by matching server 20. Sometime after visiting the Web page, matching server 20 may present Harry with an option to rate the physical attractiveness

of other users registered with matching server 20. Using display 12 and interface 16, Harry may view profiles of registered users and rank them on a scale of 1 - 9 by entering the values using interface 16; in this example, interface 16 comprises a mouse and/or a keyboard. After submitting this rating, matching server 20 will associate it with the profile and store it. Matching server 20 will also allow other users to rate profiles, thereby collecting a plurality of rankings for profiles. Matching server 20 may use this data when trying to find matches for users. One example of this is that matching server 20 may allow user 14 to specify that he/she is searching for profiles which have an average rating of 6 or above. In turn, matching server 20 may populate user's 14 result list from the pool only with profiles whose average rating is at 6 or above. Another example of how matching server 20 may use this data involves making it more likely that an entity will appear in a user's result list if the entity and that user have a similar average attractiveness rating. So, if a user has an average rating of 6, then an entity with an average rating of 5 may be more likely to appear in the user's result list than an entity with an average rating of 2.

In another example, assume that Harry is a registered user and has requested a search. While fulfilling this request, matching server 20 evaluates Sally's profile. As part of this evaluation, matching server 20 notices that Sally's profile contains feedback from other users ranking the attractiveness of Sally's profile. Matching server 20, in this example, averages that data; Sally's profile average is 6. Matching server 20 may then examine Harry's profile to determine a

similar average. If Harry's profile has an average close to 6, it will be more likely that matching server 20 will include Sally's profile in Harry's result list. If Harry's profile average is lower than 6, it will be less likely that Sally's profile will be included in Harry's result list. If Harry's profile average is greater than 6, it will be even less likely that Sally's profile will be included in Harry's result list. The more Harry's profile average deviates from that of Sally's, the less likely it will be that matching server 20 will present Sally's profile in Harry's result list.

In FIGURE 4, one embodiment is disclosed wherein matching server 20, with pool 30, may be configured to interact with another platform, such as social networking platform 50, containing a set 52 of users. Users 14 are communicatively coupled to matching server 20 and social networking platform 50. Matching server 20 may further be configured to provide users of social networking platform 50 a service by which they may search for users within set 52 or within pool 30 using the algorithms and processing of matching server 20. Matching server 20 may even further be configured to allow users of matching server 20 to search through pool 30 and set 52. Matching server 20 may be configured to parse the profiles of the entities in set 52, collecting data and applying algorithms.

In another embodiment, matching server 20 may be configured to allow users of social networking platform 50 to interact with matching server 20 using social networking platform 50. This level of integration provides the advantage of users not having to learn and sign up for a different platform.

Social networking platform 50, in one embodiment, may be a service which stores profiles of its users. This service may be further configured to provide access to the stored profiles. In one embodiment, social networking platform 50 may also allow other services to interact with users of social networking platform 50 through social networking platform 50.

In one embodiment, matching server 20 may be configured to collect requests from users of social networking platform 50 and perform a search through pool 30 and set 52. Matching server 20 may further be configured to present the results of this search from within social networking platform 50. Matching server 20 may further be configured to present entities in the search result from pool 30 as if they were entities of set 52; in one embodiment, matching server 20 may be configured to generate profiles of entities from pool 30 into set 52. Thus, users of social networking platform 50 may view all of the entities in the search result, regardless of their source (either from pool 30 or set 52), within the environment of social networking platform 50.

As an example only, consider two users: Harry, for whom matching server 20 has created a profile, and Sally, who has a profile stored in social networking platform 50. From within social networking platform 50, matching server 20 presents to Sally the ability to perform a search which Sally uses. The results of this search are presented to Sally within social networking platform 50. In this example, Harry's profile is displayed to Sally as a search result along with other entities from set 52 though Harry's profile was from pool 30. In this example,

matching server 20 uses the algorithms discussed herein and searches through the profiles stored in pool 30 and set 52. In order to display Harry's profile to Sally, matching server 20 creates a profile in set 52 using the data stored in Harry's profile in pool 30. Sally is then able to interact with this newly created profile from within social networking platform 50 in the same manner as she is other entities in set 52.

In another embodiment, matching server 20 may be configured to allow its users to interact with social networking platform 50 through matching server 20. In one embodiment, matching server 20 supplements pool 30 with set 52. In yet another embodiment, entities from set 52 appear as entities of pool 30 to the user in their list of search results. In one embodiment, matching server 20 may be configured to generate profiles within pool 30 from entities of set 52; the system may be configured to do so through capabilities provided by social networking platform 50, such as an application programming interface.

As an example only, consider two users: Harry, whose profile is stored in matching server 20, and Sally, whose profile is stored in social networking platform 50. Harry submits a search request to matching server 20. Matching server 20 may return result list 31 to Harry, which, in this example, contains an entity representing Sally's profile. Matching server 20 may accomplish this by creating profiles in pool 30 that correspond to the profiles found in set 52. Once these profiles have been imported into pool 30, matching server 20 may then search through pool 30. While doing so, matching server 20 applies the algorithms and scores discussed herein.

Thus, in this example, matching server 20 has been configured to both search and apply scoring algorithms to entities in pool 30 and set 52. Further, in one example, Harry is not able to distinguish that Sally's profile was originally stored in social networking platform 50. Rather, matching server 20 presents Sally's profile in the same manner as other profiles stored in pool 30. Thus, in this example, Harry may use favorite button 34, view button 33, and contact button 35 when interacting with Sally's profile in the same manner as described above.

One advantage present in various embodiments is that a user has a wider pool of entities to search through. Another advantage is that a user does not have to sign up with several platforms to search through the users on those platforms.

FIGURE 5 is a flowchart illustrating one embodiment of how result list 31 may be generated. At step 62, matching server 20 generates pool 30, as described above. At step 64, matching server 20 applies a filter to pool 30, removing certain entities; in various embodiments, this filter is based on user's 14 own sex and the sex user 14 desires to be matched with. At step 66, matching server 20 may be configured to apply algorithms to pool 30 that will generate a plurality of scores for each entity in pool 30. In one embodiment, these algorithms may include analyzing the text of the profiles of the entities in pool 30 to generate a readability score, determining how attractive an entity of pool 30 is, or measuring how likely it is that user 14 will contact an entity of pool 30. At step 68, matching server 20 may be configured to collect all of the scores from step 66; in

one embodiment, matching server 20 may use database 26b to store all of these scores. At step 70, matching server 20 may be configured to apply an ordering algorithm which will determine the order in which entities in result list 31 are presented to user 14. In one embodiment, this ordering algorithm is based, in part, on the scoring algorithms applied at step 66. The ordering algorithm assigns points to each entity and orders them based on these values, constructing result list 31. An embodiment of this ordering algorithm is summarized in the following table:

Condition	**Number of Points for Ordering**
Readability score 1 point higher than user	+33554432
Match result entity has expressed a preference for the user	+16777216
Match result entity has been recommended by a friend of the user	+8388608
User has viewed the details of match result entity	+2097152
Match result entity has commonality with an entity user has expressed a preference for	+1048576
Both have the same ambition	+128
Both have the same beliefs	+16384
Same answer for Build	+64
Same answer for Car	+1
Both have the same diet	+4
Both have the same preference for drinking alcohol	+131072
Same answer for Ethnicity	+1024
Same answer for Fear	+256
Same answer for Hair	+2
Same answer for Number of children	+524288
Same answer for morning	+32

Same answer for "must have"	+32768
Same answer for "night out"	+16
Same answer for "pets"	+65536
Same answer for politics	+8192
Same answer for relationship status	+0
Same answer for "romance"	+512
Same answer for smoking preferences	+262144
Same answer for sports interests	+8
Same answer for "system"	+4096

As an example only, consider a registered user, Harry, who desires to perform a search. Before processing the request, matching server 20 may ask Harry what sex he is and what sex does he desire to be matched with; in this example, Harry responds that he is a male seeking a female. After doing so, matching server 20 will generate pool 30 as described above. Next, matching server 20 will apply a filter to remove certain entities from pool 30. In this example, all males will be removed from pool 30 since Harry is seeking a female. Further, all females seeking females will be removed from pool 30 since Harry is a male. In other examples, other entities that are removed from pool 30 include entities that Harry has expressed a negative preference for before, or entities that have expressed a negative preference for Harry. After pool 30 has been filtered, matching server 20 applies a variety of scoring algorithms to the entities remaining in pool 30. These algorithms may account for various comparisons such as those based on readability, likelihood to contact, fate, and keywords described above. Matching server 20 will then tabulate these scores, storing them, in this example, in database

26b. Matching server 20 will then determine what order these entities are presented to Harry by applying an ordering algorithm. Here, matching server 20 assigns one ordering score to each entity by examining the results of the scoring algorithms. After doing so, matching server will present result list 31 to Harry, where the order of the entities that appear in the result list is based on the ordering algorithm. In this example, it is possible for result list 31 to change. Consider another user, Sally, who appears in Harry's result list. If Harry decides to add her into a separate list by using favorite button 34, Sally will be removed from result list 31 (as described above). However, Sally will also become a seed entity from which entities may be added to pool 30 (as described above). Hence, matching server 20 will update the pool, apply the filters, apply the scoring algorithms, tabulate the results, apply the ordering algorithm, and update result list 31. As another example, an entity may update their profile which can change result list 31. For example, assume Sally's profile had an ordering algorithm score that placed her within the top 20 entities in result list 31. Sally then changes her profile which results in keywords that match Harry's profile being added to her profile. Matching server 20 will then update her scoring algorithms. In this example, the change in Sally's profile and resulting increase in keyword matches with Harry's profile significantly increased her score. This was then reflected in the ordering algorithm as it was also applied to the updated profile. Afterwards, Sally's profile is now placed within the top 5 entities in result list 31.

In some embodiments, matching server 20 may be configured to receive required characteristics from user 14 regarding a match. User 14 may be allowed to specify such restrictions based upon any number of characteristics, including those described herein. For example, matching server 20 may allow user 14 to specify that entities that indicate they have children should not be displayed. In another example, user 14 may specify that only entities between the ages of 20 and 30 should be present in result list 31. In some embodiments, matching server 20 may implement these restrictions in step 64 of FIGURE 5. In other embodiments, however, matching server 20 may refuse to apply these restrictions to certain entities based on the characteristics of the entities. Any number of characteristics, including those described herein, may form the basis upon which matching server 20 decides not to apply the restrictions submitted by user 14. As an example only, matching server 20 may ignore the restrictions if the entity has a high enough attractiveness rating. In another example, though user 14 has requested that no profiles which are located more than 50 miles away should be present in result list 31, matching server 20 may include such profiles because those profiles have over 5 matching keywords, a high attractiveness rating, and have specified the same life goals as user 14. Thus, in some embodiments, matching server 20 may refuse to apply restrictions submitted by user 14 based on any combination of characteristics or algorithms.

An advantage present in many embodiments is that through taking into account various factors when scoring potential matches and using only very few strict filters,

a large amount of result entities may be returned to the user. A further advantage is that the ordering algorithm will put the most relevant search results first, saving the user time.

Although several embodiments have been illustrated and described in detail, it will be recognized that substitutions and alterations are possible without departing from the spirit and scope of the appended claims.

WHAT IS CLAIMED IS:

1. A matching method, comprising:

generating a pool of entities;

applying at least one scoring algorithm to the pool of entities;

tabulating at least one score from the at least one scoring algorithm;

applying an ordering algorithm to the at least one score; and

presenting a list of matches ordered based on the results of the ordering algorithm.

2. The method of Claim 1 wherein generating a pool comprises:

choosing at least one seed entity;

generating a score by comparing the at least one seed entity with other entities;

setting a threshold score;

adding entities to the pool whose score is above the threshold score.

3. A matching system comprising:

a first platform configured to store a first set of user profiles;

a second platform configured to store a second set of user profiles, search through both the first set and second set of user profiles, present to the second platform search results from the first set of profiles within the second platform, and present to the first platform search results from the second set of user profiles within the first platform.

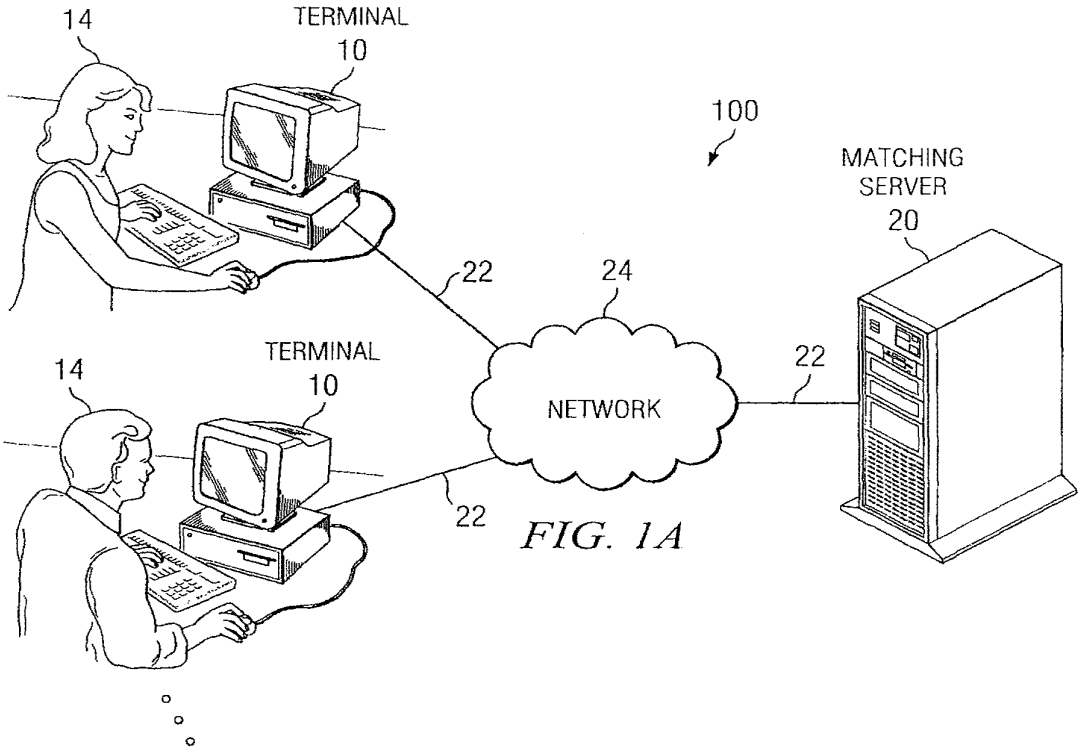

FIG. 1A

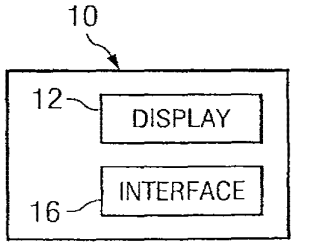

FIG. 1B

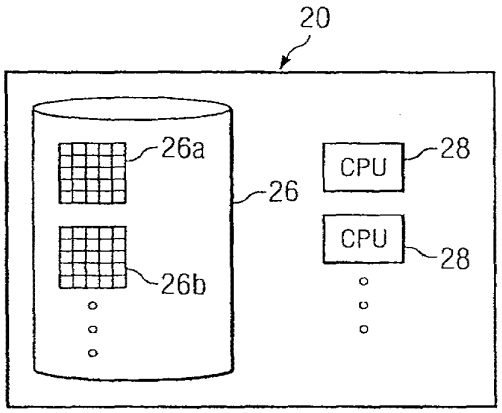

FIG. 1C

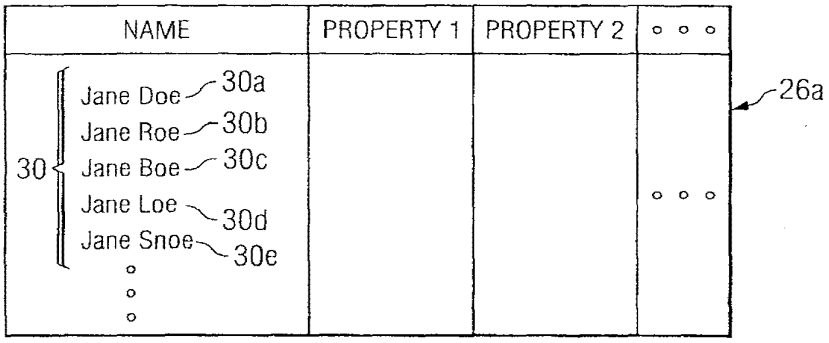

FIG. 1D

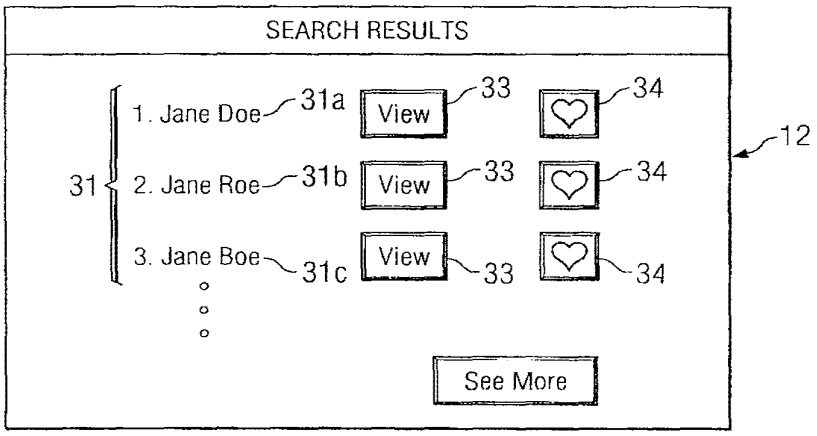

FIG. 1E

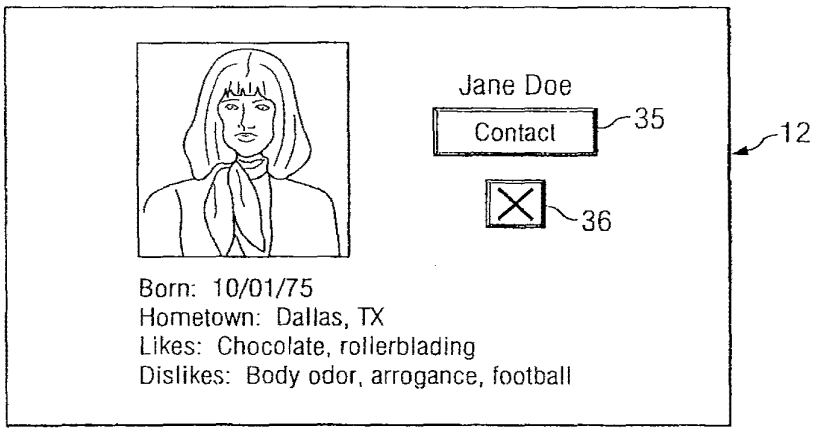

FIG. 1F

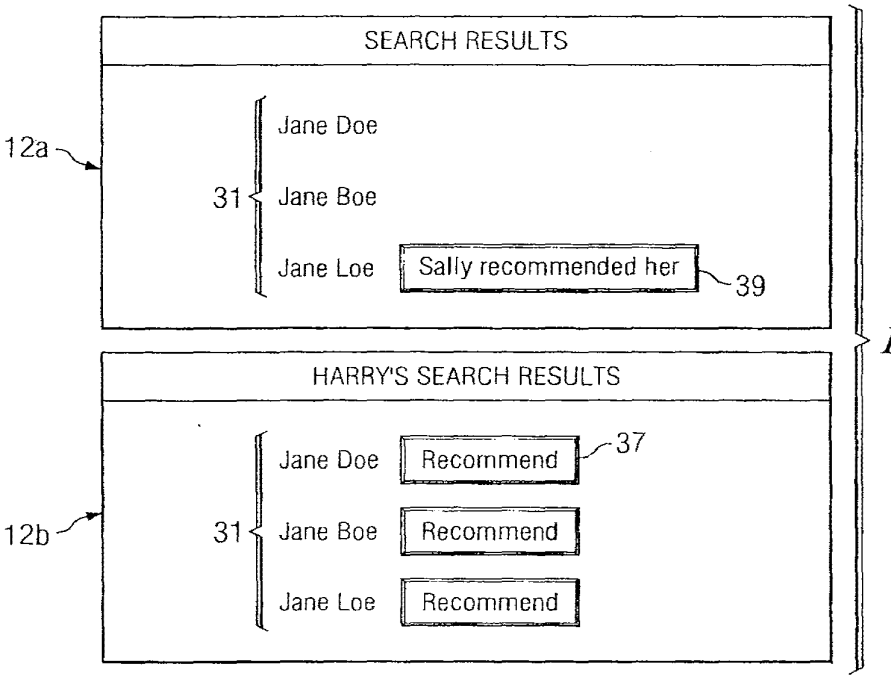
SEARCH RESULTS
12a
31
Jane Doe
Jane Boe
Jane Loe
Sally recommended her
39
HARRY'S SEARCH RESULTS
12b
31
Jane Doe
Recommend
37
Jane Boe
Recommend
Jane Loe
Recommend

FIG. 2

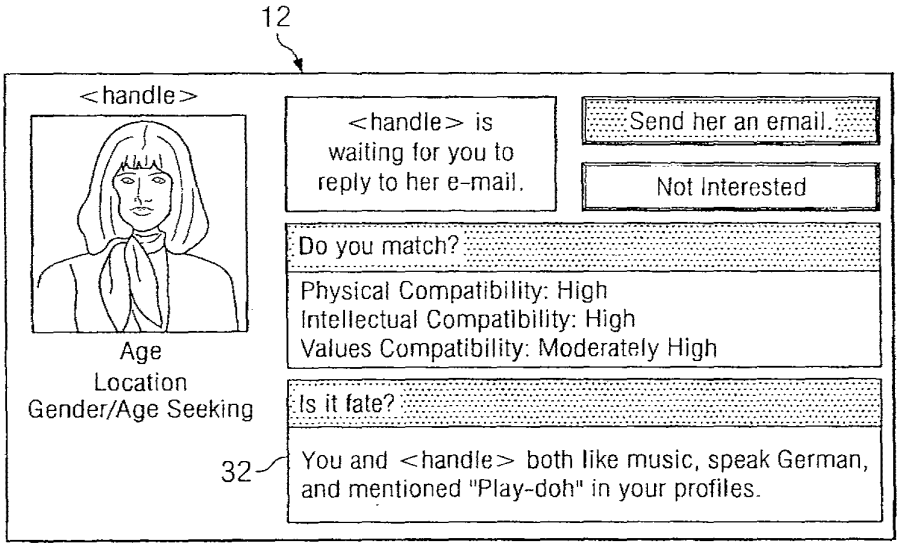
12
<handle>
<handle> is waiting for you to reply to her e-mail.
Send her an email.
Not Interested
Do you match?
Physical Compatibility: High
Intellectual Compatibility: High
Values Compatibility: Moderately High
Age
Location
Gender/Age Seeking
Is it fate?
32
You and <handle> both like music, speak German, and mentioned "Play-doh" in your profiles.

FIG. 3

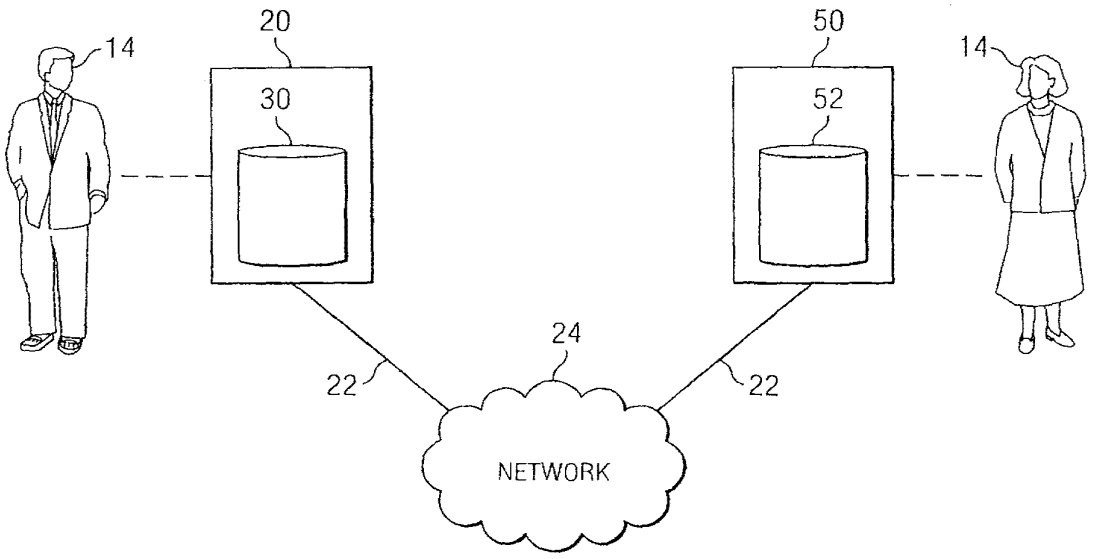
14
20
30
50
52
14
24
22
22
NETWORK

FIG. 4

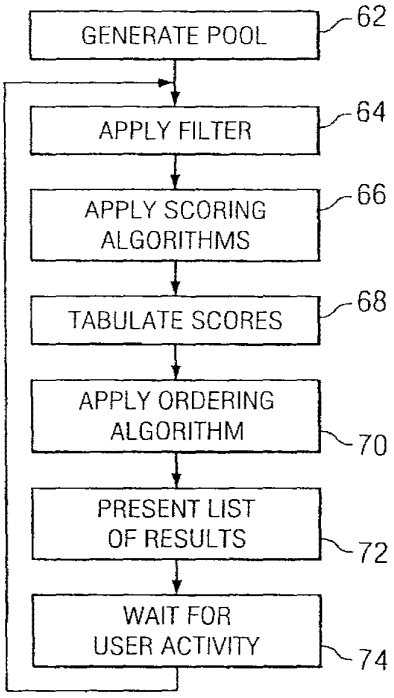
GENERATE POOL
62
APPLY FILTER
64
APPLY SCORING ALGORITHMS
66
TABULATE SCORES
68
APPLY ORDERING ALGORITHM
70
PRESENT LIST OF RESULTS
72
WAIT FOR USER ACTIVITY
74

FIG. 5